DOCUMENT RESUME

ED 243 452 IR 011 064

AUTHOR Higgins, James E.; Linville, William J.
TITLE Using Microcomputers in Schools: Some Initial Considerations.
INSTITUTION Indiana State Univ., Terre Haute. Indiana Association for Supervision and Curriculum Development.; Indiana State Univ., Terre Haute. School of Education.
PUB DATE Jan 84
NOTE 32p.
PUB TYPE Guides - Non-Classroom Use (055) -- Information Analyses (070) -- Reports - Evaluative/Feasibility (142)

EDRS PRICE MF01/PC02 Plus Postage.
DESCRIPTORS *Computer Assisted Instruction; Computer Managed Instruction; *Computer Software; Drills (Practice); *Educational Administration; Elementary Secondary Education; Inservice Teacher Education; *Microcomputers; Models; Programing; *Public Schools
IDENTIFIERS Computer Simulation; *Computer Uses in Education; Software Evaluation

ABSTRACT

Both instructional and administrative uses of microcomputers are considered in this monograph, which addresses the pressing questions of educators and administrators as indicated by requests for assistance directed to MICRONET, an Indiana school network that promotes microcomputer use in public schooling. The current condition in schools is discussed, including hardware on site, how it got there, what software is used, who uses it, and its impact. Under microcomputer uses in schools, topics include drill and practice, word processing, simulation, and tutorials. Administrative issues addressed include current practices in record keeping, inherent problems, teacher needs, degree of microcomputing skill considered essential for educators, whether to learn programming, and, if inservice is necessary, what models appear most effective. Software suggestions include both general guidelines and specific advice related to drill and practice programs, tutorials, games and simulations, and administrative uses. A selected list of software companies is appended. (LMM)

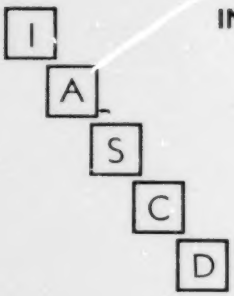
****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED243452

USING MICROCOMPUTERS IN SCHOOLS: SOME INITIAL CONSIDERATIONS

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

- This document has been reproduced as received from the person or organization originating it.
  Minor changes have been made to improve reproduction quality.
- Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.



INDIANA ASSOCIATION FOR SUPERVISION AND CURRICULUM DEVELOPMENT

James E. Higgins
William J. Linville

And

IR011064

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Dorothea Nelson

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

CURRICULUM RESEARCH AND DEVELOPMENT CENTER

SCHOOL OF EDUCATION, INDIANA STATE UNIVERSITY ● TERRE HAUTE

INDIANA ASSOCIATION FOR SUPERVISION AND CURRICULUM DEVELOPMENT

The Indiana ASCD was first informally organized in 1923. Since 1965 the association has been based at Indiana State University. Today, membership in the association approaches 600, with almost one-half of that number affiliated with the national organization. The membership of the association includes curriculum directors, supervisors, content specialists, superintendents and assistant superintendents, principals, college professors, consultants, commercial people, elementary and secondary teachers, and students. We are particularly proud that our membership represents a diversity of professionals in all areas of education. Among the association's goals is to sustain professional development that focuses on perceptive, competent curriculum planning and supervision as well as the improvement of instructional practices for children. There are five regional divisions of IASCD which hold annual spring conferences.

Marjorie Jackson
President

William L. Walker
Executive Secretary

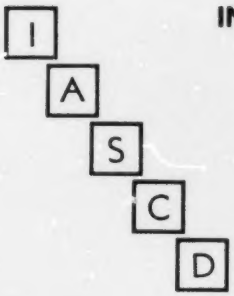
THE CURRICULUM RESEARCH AND DEVELOPMENT CENTER

The Curriculum Research and Development Center of Indiana State University provides school systems the opportunity to secure aid, encouragement, and cooperation in curriculum development projects. It coordinates the participation of University personnel engaged in curriculum work, provides information concerning curriculum development, and initiates and sponsors curriculum research projects. It is the contact point where public schools initiate inquiries regarding curriculum and acts as a vehicle for communication between elementary and secondary schools and the University. Although the CRDC operates as an agency of the School of Education, it represents all departments of the University engaged in curriculum development projects.

J. Stephen Hazlett
Dean, School of Education

James E. Higgins
Director

USING MICROCOMPUTERS IN SCHOOLS: SOME INITIAL CONSIDERATIONS



INDIANA ASSOCIATION FOR SUPERVISION AND CURRICULUM DEVELOPMENT

James E. Higgins
William J. Linville

And

CURRICULUM RESEARCH AND DEVELOPMENT CENTER

SCHOOL OF EDUCATION, INDIANA STATE UNIVERSITY ● TERRE HAUTE

Copy for this monograph was produced entirely with the use of microcomputers, word processing programs, and printers, and was authored by:

James E. Higgins
Professor of Elementary Education
Indiana State University

William J. Linville
Professor of Elementary Education
Indiana State University

School of Education Publications:

Dorothea Nelson
Managing Editor

Carole Gustafson
Copy Editor

Sue Lightfoot
Technical Assistant

Published by:

Indiana Association for Supervision
and Curriculum Development
and
Curriculum Research and
Development Center
School of Education
Indiana State University
Terre Haute, Indiana 47809

January 1984

TABLE OF CONTENTS

INTRODUCTION 1

THE PRESENT CONDITION 3

CLASSROOM USES FOR MICROCOMPUTERS 7

THE NEEDS OF TEACHERS 11

SOFTWARE SELECTION 17

CLOSING COMMENTS 25

APPENDIX 27

INTRODUCTION

This monograph is an attempt to familiarize the reader with information appropriate to the using of microcomputers in the public schools, for instructional and administrative purposes. As a publication sponsored jointly by ISU's Curriculum Research and Development Center and the Indiana Association for Supervision and Curriculum Development, the reader might assume the contents are to be directed toward Indiana schools. Such is not the case. There is reason to believe that Indiana's situation closely parallels those which are commonplace in most other states when it comes to this topic. Efforts are being made to meet the challenges. It is suggested that if Indiana does indeed differ from her neighboring states it is that she has the potential to lead in microcomputing efforts, due mainly to some recent legislation directed at developing educators' skills with the microcomputer. How that legislation is actually carried out will have much to say in terms of where Indiana stands. At the time of this writing, however, the powers vested with the decisions of how best to proceed appear to lack any clear direction of purpose.

Technology is progressing at such a rapid pace that the authors of this monograph fully expect the contents to be somewhat outdated by the time the ink has dried. If such is the case, the reader is encouraged to consider this treatise from a historical viewpoint. However, because educational institutions, as opposed to the business community, tend to proceed in less rapid fashion when embracing new developments, this concern of literary obsolescence may be greatly exaggerated. Such is likely far more a blessing than a curse, however.

The authors come from a background rich in educational experiences as they relate to the microcomputing revolution in the schools. These experiences have been gained in a rather limited period of time, but their intensity has been great. After all, is there any previous "intruder" to our educational institutions that has made as much noise as has the microcomputer? The authors are deeply involved, also, in the ongoing efforts of ISU MICRONET.

MICRONET is a network of schools representing about 60% of the Indiana school corporations, developed in 1983 with the primary purpose of promoting the use of microcomputers in public schooling. The network is headquartered in the School of Education at Indiana State University and is a branch of the Curriculum Research and Development Center. Membership is through a modest subscription fee and entitles the member schools to receive oft-published newsletters, free registration to conferences, and software examination privileges at ISU MICRONET sponsored events.

In the realm of educational computing, there are surely more questions than there are answers. The following chapters have been written to address the most pressing questions as

indicated by the requests for assistance received at MICRONET headquarters.

The current condition in schools is discussed. What hardware is on site? How did it get there? What software is present to drive it? In general, are there any staff on duty who care to drive it? And, what difference does all this make?

Another chapter is devoted to microcomputer uses in schools. What about drill and practice? Is wordprocessing worth looking into? Simulation? How about tutorial? Also, there is the administrative side to the micro in record keeping. What are the current practices? Are there inherent problems?

What are the needs of teachers? How much microcomputing skill is considered essential for every educator? Do teachers need to learn programming? If inservice is necessary, what models appear to work better than others? A chapter is set aside for this.

Finally, a rather detailed chapter concerns itself with possibly the most important of all decisions to be made at the local level--software selection. Without software, hardware is virtually useless. The cost of commercial software mandates careful consideration of numerous factors.

The time appears to be now. This technological revolution gives every indication of increasing in magnitude rather than fading, a la some previous educational innovations. Educators in higher education and in the public and private elementary and secondary schools have the responsibility for using the microcomputer to the advantage of all. This monograph is dedicated to making a positive contribution.

Chapter One

THE PRESENT CONDITION

It is most curious that a monograph concerning educational computing would contain an entire chapter on the current state of affairs. The present condition is most fluid. However, it becomes necessary to attempt gaining some perspective as to where schools are before ever considering where they might direct their future efforts. Thus, this is a rather extensive look at what is known and/or perceived to be.

When ISU MICRONET was accepting memberships during the summer and fall of 1983, part of the membership application completed by public school personnel addressed the current resources, both technological and human. This chapter is based in that data, unless otherwise noted, and refers most specifically to Indiana schools.

Hardware

Hardware is in place in the vast majority of school systems, although not all individual schools have access to it at all times, as when the computers are placed in the building for a period of 3 to 6 weeks and then transferred to some other school. The hardware referred to here consists of computers, monitors, and disk or tape drives rather than peripherals such as printers and graphics tablets. The hardware has been purchased through funds that were budgeted for computer purchases, from cumulative building funds, and through monies set aside for other uses, including Chapter funds. Numerous elementary schools have benefitted from the efforts of parent/ teacher clubs holding bake sales and the like to finance hardware purchases.

In general, three brands of hardware control the school market: Apple, Radio Shack, and Commodore. At least 50 percent of the schools have Apple machines, and that percentage appears to be increasing. Radio Shack was the first brand introduced to the schools, with placement tending to be in secondary schools. Radio Shack's current share of something less than 30 per cent is due primarily to those early sales. The Commodore company placed some PET machines in schools, especially when they had a "buy two, get one free" offer. Since then, however, that company's primary seller has been the Commodore 64, and current sales to schools may be considered moderate.

The remaining machines with a few placements in schools are Texas Instruments, Atari, Timex/Sinclair, and Franklin. IBM may make inroads into the education market, but nothing significant is happening at the time of this writing. Because IBM is IBM, many feel it is only a matter of time, but the cost of the machine may be a factor against this happening. A few schools, and very few to be sure, use mini-computers (small mainframe computers) linked to terminals, rather than

the stand-alone micros. The cost of the mini generally is in six figures, making such a purchase quite impractical for instructional use, even if there were educational software available for these machines, which there is not.

Software

Software, the programs needed to drive the hardware, is yet another story. The hardware purchases in the majority of schools required such a large capital expenditure that little was left to purchase the software. Whereas this is construed by many to be poor financial planning, the decision to purchase hardware had to be made first, and an adequate number of machines had to be purchased in order to make any difference at all. The ask of software selection that lies ahead for most schools is a crucial one. There is a great deal of quality educational software being produced. The key decisions to be made at the local level will be in determining which software to buy. These decisions will likely be difficult ones to arrive at properly.

Software has been an issue for the hardware producers, too. The "shakeout" through the summer and fall of 1983 resulted in Texas Instruments' announcing they were discontinuing production of the TI 99/4A. Rumors persisted that Atari appeared to be getting closer to discontinuing hardware production, especially if their large financial losses continued. Still, Atari was introducing a new model. Part of the reason centered around fierce competition when Commodore became quite aggressive in going after the home market. There would also appear to have been poor decisions concerning software development. In essence, the companies whose machines were being backed by third-party software development were surviving; those choosing to create their own were not. In this realm, Apple is the undisputed leader. It is estimated that 99 of every 100 pieces of educational software are produced to run on the Apple machines. Commodore is benefitting from software development, but the quantity (individual software packages from which to choose), although increasing, is far less than for Apple. Tandy (Radio Shack's parent company) was somewhat slow in encouraging other companies to produce TRS 80 software, preferring to attempt to develop all their own. Quite a bit was actually developed, but not nearly enough to satisfy the need, and Tandy elected to market it less efficiently than most other companies.

Just as there were business failures due, in part, to lack of software development, so can the educational computing movement die if there is lack of appropriate software at the school level. In fact, this appears to be the only way of stopping the technology from becoming firmly entrenched in the educational curriculum--through ignoring or neglecting the software issue.

As an aside, all software does not have to be purchased. According to data collected by ISU MICRONET, a large number of schools currently employ at least one teacher who possesses

adequate skills to write fairly simple, though useful, educational software. The numbers of these teachers are steadily increasing, and administrators responsible for hiring personnel are just beginning to see the advantages of seeking someone with these skills.

To repeat, there is quality software presently available to schools. It is not inexpensive, to be sure, but most of it is reasonably priced ($20-$50). It is important to examine software before buying, however. ISU MICRONET is supported through donations of software by software companies considered to be the leaders in the educational marketplace. (Their names and addresses are located in the Appendix.) During the 1983-84 academic year, numerous groups of teachers and administrators from MICRONET member schools have visited MICRONET for the purpose of examining this software.

For those in need of assistance concerning software selection, Chapter 4 of this monograph will prove to be most helpful.

Chapter Two

CLASSROOM USES FOR COMPUTERS

Increasing numbers of microcomputers are being purchased for use in public schools and may be found at all levels. It is ironic, perhaps, that many individual schools and entire school corporations joined in the rush to purchase the hardware and, only after that, began to give really serious consideration to the questions which arose with respect to how best to use the equipment. In other instances, even though very real and serious consideration had been given to potential uses prior to computer purchases, individual teachers are still searching for helpful suggestions regarding how best to use the computer as an integral part of an ongoing curriculum.

At the present, it is highly likely that the single most prevalent use that is made of the computer in the classroom is for computer-assisted instruction. Even this one aspect of computer usage, however, can involve a variety of considerations and potentials. One such consideration, for example, involves exploring the potential of the computer as a tool to be used for drill and practice.

It is unfortunate that the drill-and-practice use of the computer has come to have such a distasteful connotation in so many circles in such a short period of time. This has probably been fostered by experiences of educators with early types of software in which drill-and-practice programs appeared to be nothing more than "electronic worksheets." Even if this criticism were justified, the argument might still be presented that the immediate reinforcement the student receives makes the "electronic worksheet" superior to the printed one. More recently, even the drill-and-practice programs have begun to illustrate the benefits of creative and imaginative programming. Many current examples of software in this category are skillfully constructed and are able to capture the interest of the student.

Every skillful teacher recognizes the need for some form of drill and practice following the development of concepts and the skills which are attendant to the concepts. These same teachers also realize that few skills are mastered by students after only one presentation by the teacher, nor are the skills mastered by all students in the same way or at the same time. It is because of this that teachers find that more students, particularly those of average and below-average abilities, need their individual assistance to foster the development of necessary skills. The unfortunate fact the teacher must face, though, is that more students are in need of individualized help than the teacher has time to provide. It is for these students, perhaps, that one real potential of the computer as a tool can be realized: to assist in providing individualized drill and practice for students who need the extra practice yet cannot get the help from the teacher because of time constraints.

Using well-written software, drill and practice can be provided for individual students when it is needed, covering only those skills that need to be practiced. The computer can exhibit infinite patience and repeat a drill-and-practice sequence as many times as necessary until a predetermined level of mastery has been attained. The added potential of presenting the drill in a variety of formats can also serve to lessen the possibility of student boredom that can often be the result of workbook pages which all appear to be mere repetitions of what has been done previously.

Coincidental to the use of drill-and-practice software, or in addition to the use of that type of software, teachers at all levels are finding increasing uses for programs that are more tutorial in nature. Rather than just drilling a student on a skill that has already been presented, the tutorial programs may actually present and develop concepts in much the same way that the classroom teacher might introduce new topics. Realizing that this type of software cannot, nor should not, take the place of the classroom teacher, several potentials are released when tutorial programs are used.

Many instances arise daily in the classroom in which the teacher senses the need for a parallel means to help students understand the essential concepts embodied in a new topic that is being considered. Often, internal or external pressure to "cover" a specified amount of material causes the classroom teacher to move on to other areas without spending the time to examine the new concepts in different ways. Well-designed tutorial programs, used with individual students or with small groups, can become effective supplements to the teacher's initial presentations of these new concepts. This additional exposure to the concepts, whether it be in a form parallel to that presented by the teacher or a different approach to the same ideas, can often spell the difference between understanding and not understanding on the part of the students involved.

A valuable side benefit can also be realized when the teacher uses carefully constructed tutorial packages. Newer tutorials, and some drill-and-practice programs also, permit considerable degrees of record-keeping capabilities for the teacher. One of the more difficult and time-consuming aspects of attempts to individualize classroom instruction involves the need to keep track of each student's progress. For the teacher who is working daily with classes of 25-30 students, especially with multiple sections, the record-keeping task becomes an almost insurmountable one. When the parameters of the tutorial software package permit the saving of each student's progress each time the package is used, much of the teacher's time is freed for other, more important tasks. In fact, with many tutorial programs, not only does the teacher have the option to review the progress of individual students, but assignments for individuals may be made within the parameters of the package. Then, based upon the student's performance on a given sequence of instruction compared to a predetermined mastery level, the teacher may decide that the

student may skip some program segments, repeat some, or leave the package to move on to something else.

Although this discussion for both drill-and-practice and tutorial programs has presumed that both were a part of a computer-assisted instructional approach, when the software permits the teacher to keep accurate records of student progress, assignments, and other aspects of daily class work, then the approach is leaning more toward one that is computer-managed instruction.

While most of this discussion of the uses of tutorial software has dealt with using the computer to present or review concepts that parallel what the teacher is doing on a day-to-day basis, there is still another potential that should not be ignored. In many classrooms, some students have the ability and background to master concepts and necessary skills at a far more rapid rate than many of their classmates. These students present a different type of challenge for their teachers, for unless they are provided with some type of supplementary instruction that can challenge their abilities, they become bored and view the class as a "turn-off." As is the case when attempting to provide appropriate experiences for the below-average student, most classroom teachers find that the time is simply not available to plan separate sets of learning experiences for the more academically gifted students. Here, also, well-designed tutorial software can be used to provide the "extra" educational challenge for these students.

Perhaps one of the more promising areas of computer software for the above-average student is an area that has only recently begun to receive considerable attention by software developers. This is a category of software that includes simulations and problem solving. Many such packages require the student to "role play" in a situation that requires the development of strategies to solve problems that are unique. The problematic situations may utilize previously developed concepts and skills, yet have solutions that are totally dependent upon student input. While most "problems" that are found in the exercise and/or activity pages of textbooks have a predetermined solution and may only be solved in one way, simulation software may permit an almost limitless variety of "endings" to the problem, dependent upon the student's choices of means to attack the problem. Usually, in both simulations and problem-solving types of programs, the student who exhibits the better problem-solving strategies is "rewarded" by a more profitable or more pleasant outcome than is the student who has done a poorer job of gathering data, analyzing, synthesizing, and hypothesizing. Many educators would argue that it is the ability to solve unique problematic situations that distinguishes real learning from superficial learning. For the teacher who wants to provide such experiences for students, increasing numbers of software packages from this category are becoming available. It is also worth noting that the programs of this type that are appearing on the market are designed not only for older

students. Increasing numbers of these packages are actually designed for students in the elementary grades.

While not usually thought of in the sense of an "instructional package" as might be true for drill-and-practice or tutorial programs, still another type of software is being increasingly used in school applications. Word processing programs are being used by teachers at all levels to encourage and stimulate students' creative writing endeavors. Some of the newer word processing programs have sacrificed some degree of power in the interest of simplicity, enabling even elementary school students to use the packages effectively. In the past, many teachers have indicated that one of the barriers to getting students to do more writing is the fact that students do not recognize the need for editing and rewriting parts of what they have written. Additionally, the mechanics of rewriting, which in many cases actually involve redoing a paper completely, have discouraged many students. Since word processing programs allow corrections and changes in a written document to be made so easily, many teachers are finding that their students are much more receptive to writing assignments when the word processor is available.

Finally, although not considered as a tool to be used in instruction per se, the use of data-base programs for keeping student records should not be forgotten as a most viable use of the computer in education. Increasing numbers of these programs are easy to use and permit the storing and/or retrieving of information in a fraction of the time required by more conventional methods.

Most schools now have microcomputers. As illustrated here, the uses to which the computers can be put within the school setting are many. In reality, perhaps the only real determiners to how educators may most effectively use the computer are the availability of appropriate software and the imagination and creativity of the teachers and administrators who will be using the computers. The computer can become a tool to help educators realize some additional educational goals, or it can be relegated to the corner of the room or a closet to gather dust because no one wants to bother to learn how to use it. Many educators remember the rush to purchase science equipment for schools in the post-Sputnik era. Today, much of that equipment sits unused in schools, gathering dust. Educators must not permit microcomputers to be the dust catchers of tomorrow.

Chapter Three

THE NEEDS OF TEACHERS

There has recently been a great clamor around the country warning all within shouting distance that the youth of America must become computer literate if they are to be competitive members of the nation's workforce. Even more recently, the original clamorers have been joined by others who have decided that American youth can only become computer literate if their teachers are also brought into the literacy ranks. These two needs, as perceived by many in society, have raised some interesting questions and created a variety of problems. For example, what does the term "computer literacy" mean? Does it mean the same thing for students that it does for teachers? Are there "degrees" of literacy? Can some groups get by if they are just a little literate, while other groups need to be more literate? It is unfortunate, but true, that no real answers exist for these questions at present. Advocates of many different curricular and inservice patterns may be found. It is our purpose in this monograph to suggest some considerations which we feel should be kept in mind when decisions are made with respect to the needs of teachers in the whole area of computer use.

In the first place, the needs of all teachers with respect to computer literacy (whatever that means) cannot be the same. To a great degree, the kinds of uses that the individual teachers may find for computers in their own teaching will determine how proficient those teachers need to be. Obviously, the person who is charged with the responsibility for teaching a class in programming to junior high or senior high students will have a greater need for in-depth proficiency than will the teacher who will use the computer only for computer-assisted instruction. Let us consider first what might be construed as a minimal kind of literacy that all teachers should have as we prepare for greater computer use in classrooms.

All teachers need to understand what the various pieces of computer equipment are and how they function as a unit. While it may seem simplistic to suggest, this includes the following items:

1. The computer, itself, has two kinds of memory. One part is permanent within the computer. The other part of the computer's memory may be changed, and it is in this part that "instructions" are temporarily stored which tell the computer what to do.

2. The monitor is a peripheral device that looks like a small television set. It is on the monitor's screen that print or graphics from the computer's program appear.

3. The disk drive is another peripheral that enables the computer to send instructions from a floppy disk to the computer or from the computer to a floppy disk.

4. Floppy disks are a form of magnetic storage medium. It is on these disks that instructions (programs) to be used by the computer are stored.

For many teachers, then, information of this type will form the basis of their needs with respect to computer literacy. Beyond this, teachers must learn how to handle and generally care for the equipment. This is especially critical with respect to the disks since they can be subject to several kinds of damage. Teachers also must receive inservice assistance to learn how to properly insert a disk into the disk drive, get the equipment turned on, and get a piece of educational software started. While it may appear obvious, attention must also be given to how the teacher stops a program that is being used.

In addition to learning how to use the computer equipment in a physical sense, teachers also must receive assistance with respect to techniques for evaluating and selecting educational software. Inservice help should address itself to helping teachers realize the types of software available, the potential uses by type, and criteria for the wise selection of software.

Finally, many teachers already perceive that they need assistance with respect to how they can implement the computer into their ongoing curriculum. Most elementary teachers have, for some time, used learning centers in their classrooms to provide enrichment or remediation for individuals or small groups of students. With a limited number of computers, this same type of technique will need to be developed by teachers at all levels who want to make use of computers but realize that funds do not permit a ratio of one computer to one student. Part of the inservice work in this area will need to be designed to help teachers realize that it is not necessarily mandatory, nor even desirable, that each student have his or her "turn" at the computer each day. This reference is intended, of course, for those classes in which the computer is used as an instructinal tool, not where students are actively engaged in learning to program. In that case, each student must use the computer daily.

If the items that have been mentioned are to be construed as a minimal or level-one type of computer literacy for teachers, then what kinds of background experiences would teachers operating at a higher level need? In addition to the knowledge already discussed, these teachers must develop some degree of proficiency with respect to programming skills. This will, of course, entail the need to learn one or more computer languages. Without attempting to enter the debate regarding which computer languages are most desirable, at the present time it would appear that teachers who wish to equip themselves beyond the minimal level of competency should learn BASIC, LOGO, and/or PILOT.

Many would argue that teachers do not need to know how to program in any computer language to be able to use computers

in their classrooms. If one is to assume that the only use for computers in classrooms will be for computer-assisted and/or computer-managed software packages that have been commercially developed, then the assumption might have merit. The teacher who has some programming skills, however, will be able to construct or help others to construct software that is consistent with specific, local instructional objectives. Also, these teachers will be able to use educational program listings found in computer journals and/or professional journals and modify the programs for specific classroom needs.

It is patently obvious that *every* teacher does not have the desire or the need to learn to program. However, each school corporation needs such persons. Ideally, one or two teachers per building could do much if they served as resource persons to assist their colleagues in constructing simple programs, or if they were provided with enough released time to do program construction for other teachers in the building or system.

Whether the consideration involves getting teachers prepared at the most elementary level of computer literacy or at the level that requires programming skills, there are several questions which arise with respect to developing the necessary competencies. Perhaps the most pressing of these questions concerns how teachers are to acquire the necessary skills and concepts to become "literate."

If the goal is to develop computer literacy for teachers at the minimal level, then a variety of inservice models might be considered. A number of school corporations have attempted to provide assistance for their teachers by having a "one-shot" session. It would be easy to generalize that these attempts to acclimate faculties to the instructional uses and implications of computers are unsuccessful. If, however, the major purpose of such types of inservice is purely to help teachers develop some general familiarity with computers, then the inservice can be successful.

On the other hand, when the perceived need is to develop some genuine competencies among faculties with respect to how to use the disk operating system; how to load, save, and delete programs using a disk drive; and how to load and use commercial or public-domain software, then the "one-shot" session is not sufficient. At least two, and preferably three, sessions of at least one hour each are needed to develop these kinds of skills. Not only are more sessions needed, but they should be provided with enough intervening time for the inservice participants to "digest" and practice the skills being developed.

For individual teachers, or groups of teachers, who wish to develop competencies beyond the minimal level, a much more rigorous program is needed. If the competencies are to be developed in an inservice atmosphere in the local school setting, a minimum of 30 hours of inservice instructional time would be needed for the participants. If possible, 45 hours would provide a much better opportunity to really develop some

of the desired competencies. If the preparation is to be done within the context of enrolling in classes at a college or university, a class carrying three semester hours of credit should be the choice of those who wish to develop some programming skills.

Attention must also be given to the choices which will be made concerning who will conduct the inservice training or classes. Past experience has indicated that those who do the instructing *must* be professional educators themselves. In most instances, it would be better if the individuals who carry out the training processes are not full-time computer programmers, nor should they be persons who are in the business of training computer science majors for industry. While these individuals are likely to be highly competent in their own work, it can be far too easy for them to forget that the needs of teachers are quite different from those of computer science majors.

It is more than a little ironic that the microcomputer, while offering such tremendous promise for education, can also be the most potentially intimidating device ever brought into the classroom. Many teachers are already quite apprehensive about having to work with computers. Inservice experiences led by persons who do not understand this apprehension and who do not try to structure meaningful and understandable experiences for the inservice participants will find that the teachers are "turned off." It is very unfortunate that those teachers who are turned off will, in all likelihood, never be willing to go through another similar experience.

In passing, it should be noted again that the needs of teachers who are going to be actively involved in teaching computer programming to students in their own classrooms are obviously quite different from the needs of the vast majority of teachers with respect to literacy. The suggestions contained in this monograph should not be construed as being intended for the former group.

In the final analysis, the impetus to direct a program to assist teachers in developing competencies in educational computer uses must come from somewhere. At the present, this impetus has been furnished by individual teachers who have perceived a need and taken workshops or classes on their own. These classes have often been taken at colleges and universities where faculties are understaffed and underequipped to handle the needs of all teachers. In other instances, individual schools or school corporations have undertaken the responsibility of contracting with some individual or individuals to provide inservice training for teachers. In Indiana, much has been verbalized regarding the intent to make all Indiana teachers "computer literate." At present, no plan to implement such an ambitious undertaking has been adopted.

In summary, the need is obviously and painfully clear. The role of the computer in educational settings does not appear to be another passing fad. The uses to which the

computer can be put in classrooms appear to be limited primarily by the imaginations of professional educators. However, if assistance is not provided, and quickly, for teachers to develop some proficiencies with computers, then a golden moment may be lost.

Chapter Four

SOFTWARE SELECTION

In the world of educational computing there is no single area of decision making that is as important as the determination of what software to purchase. The "glory" decisions are in selecting hardware. The "nitty-gritty" decisions concern software.

In essence, almost anyone has the ability to select hardware. From a technical standpoint, practically all brands are of good quality. That includes most of those that have fallen by the wayside in the highly competitive high-tech marketplace. Those educators who opted for Atari machines over Apple did not make poor hardware selections. Nor would they have if the decision had been the other way around, or if they had selected Texas Instruments, Commodore, or Radio Shack. The major objective was to introduce students in the schools to computers, and that objective was met.

The "how" of initiating computing into the curriculum is not a hardware matter as much as it is a matter of software. For what does the school plan to use computers? Will the elementary schools provide drill and practice activities? Will special education teachers use the machines to develop IEPs? Maybe the counselors wish to provide some occupational guidance via the computer. Central administration might see how their record keeping could be streamlined, thus saving both physical space needed for storage and also saving time in information retrieval. The high school business department may choose to teach word processing. The uses of computers in schools are limited not so much in the capabilities of the computers or the software that is currently available; the limitations reside in the creative imaginations of the people who use them. In every case, however, appropriate software is a requirement. In no case can the computer meet an objective without software driving it. The proper software must be selected, and the selection process is best carried out by people who will be most directly affected by the decisions made.

This chapter is devoted to suggestions that may assist the reader to attain success in the software selection process. The primary emphasis is placed on software for classroom use as a tool in the instructional process, but some attention is given to other uses. The authors base their comments on their experiences with the various software categories, but specific software is not mentioned. To identify software by name would serve to promote some and denigrate others, and what is good in the eyes of one evaluator is not necessarily so in the eyes of another.

Not all software is developed for use with all brands of hardware. Good sources of information as to what software is available to run on what machine are the software publishing houses themselves. The reader is encouraged to write to them

requesting product catalogs. A partial listing of publishers is found in the Appendix of this monograph.

General Guidelines

Once the objectives are known, the process of review and selection begins. Never buy software sight unseen. A car is purchased after a test drive; require the same thing prior to software purchase. Consider the following during the test drive:

1. There should be documentation (written instructions and explanations supplied by the publisher) that is understandable. This is not to say that the software is used by merely inserting the disk into the drive and typing RUN. Some software packages are extremely complex, yet they become easily usable because of the clear instructions provided by the publisher. Sometimes a specific piece of software gets a "bad rap" because those examining it fail to adequately read the documentation that accompanies it. This can result in missing out on a program that would satisfy a real classroom need.

2. Once the program is running, directions and information appearing on the monitor screen must be clear to the user. Any questions appearing that require a response from the person sitting at the keyboard should be specific. If this is not the case, go looking for another piece of software to meet your needs.

3. Graphics (pictures) are important to most users. Some programs contain more graphics than anything else, and those responsible for software selection need to take care not to confuse educational quality with sensory stimulation. If the graphics are appropriate to the educational content of the program, those graphics are likely an enhancement to the program. Programs with no graphics at all, however, may still have good educational application.

4. All words output to the screen must be spelled correctly (unless the program is a spelling program where words are purposely misspelled) and all sentences should be grammatically correct. If this is not the case, assume the producer of the software lacked a real commitment to quality.

5. Don't lose sight of what you are looking to find. Some software, at first glance, appears to be "just what the doctor ordered." This appearance is enhanced by a clever use of graphics and possibly the inclusion of sound effects. Only when the software selection committee focuses attention on the actual objectives being addressed by the software under study will it be determined whether a particular piece of software is good or not so good.

6. Error traps must be included in the program. An error trap is a provision for handling any kind of response entered at the keyboard, for instance, if a child is asked to enter a number word ("two," for example) and types "2" instead. In some poorly constructed programs, this action will result in real problems. The program might even crash (stop altogether). If error traps are missing, the software should not be considered for purchase.

Drill and Practice

Much software exists for assisting the teacher to provide drill and practice. Some critics have suggested drill and practice is a poor reason for investing thousands of dollars in hardware and software. They suggest drill and practice occupies a "low station" in the classroom when compared to simulation and administrative functions. Such thinking is best described as illustrative of an elitist point of view. The other side of the argument presents drill and practice activities at the computer as a way of freeing the teacher to spend time in "educating" the students. Regardless of all this debate, some students require a great deal of drill and practice before they are able to master a skill. The computer may be of great help if the software utilized meets the following criteria:

1. The program must be interactive. It must require the student to frequently enter responses. It is a cliche that learning takes place best when the learner is actively involved, yet it remains true. It should be obvious that the involvement should be mental as well as physical.

2. The program must employ motivational techniques to reinforce learning and also to entertain. Time on task has been shown to play a key role in the learning of basic skills. If the program lacks the necessary stimuli, such as color graphics and/or sound, the pupil will likely become bored with the task, thus the desired learning will fall short of the mark.

3. Pace in learning is important. If the learner has no control over the pace of the program, the program is less than ideal. As an example, directions and other text on the screen should remain on the screen until the user wishes to continue. If a built-in "timer" moves all users along at the same pace, the program fails to individualize as it should.

4. If the content to be practiced is best achieved through acquiring skills in stages or levels, the computer program should reflect this. For instance, math facts in multiplication might be learned in stages--the 2s and 3s before the 9s. The software should allow the user to determine the level he/she wishes to address.

5. Some form of reinforcement must be provided for correct responses. This reinforcement should not "overshadow" the content under study, however. For instance, if a 30-second graphics display reinforces each correct answer to the math facts discussed above, tedium as well as "overkill" will be the result.

6. If the software offers the teacher the opportunity to modify the program's content to better meet the specific needs in a particular classroom, this is an indicator of quality. As an example, a spelling program would not be too practical unless the words could be changed from one use to the next.

7. The program should offer record-keeping capabilities, thus allowing the teacher to keep track of the progress being made by each student. This record keeping referred to here is in the program itself. Once the student is finished, a "report" of his work is added to the program disk by the computer. At the end of the school day, the teacher enters a password at the keyboard and is provided with information on all students who have interacted with the program that day.

Tutorial

Although drill-and-practice software is directed toward skill development, most specifically basic skills, good tutorial software is used in the formulation of conceptual understandings. The content will include skill utilization, the "how" of doing things, but will also proceed to greater levels of difficulty, providing activities that question "why" something is so. Program attributes that provide high pupil motivation and frequent pupil involvement remain every bit as important with tutorial software as with drill and practice software. Other considerations are as follows:

1. Variety is important. It is likely that some students will fail to learn if the material is presented in a single fashion. While the program is running, the student will be asked to enter responses to various tasks presented. If those responses are indicative of less than adequate learning, additional tutoring is necessary. If that instruction is nothing more than a repeat of the initial "lesson" then individual differences in students have been ignored.

2. The content must be logically sequenced. Since the learning is taking place at the computer open ended discussions are not possible as they would be in a traditional classroom environment. If the content is not sequenced properly, the "cart before the horse" effect will result. The teachers who are to use the software being considered will serve as the best resource people concerning this criterion.

3. Branching must be conditional. In a traditional classroom setting, the teacher will "branch" his/her teaching depending on pupil responses to questions under study. A good tutorial program will do the same thing. If correct answers are registered by the student, the program advances to the next level of learning. If learning is incomplete, the program branches to a remedial/review lesson.

4. Reinforcement is a must. When the student's responses indicate learning is taking place, positive statements to that effect should be output to the screen. When incorrect responses are given, the student should be made aware of these, but in a non-derogatory manner. (There are a few programs available which seriously violate this criterion.) Also, if graphics following incorrect responses are too "spectacular," students may purposely miss a question in order to "see the show."

5. Frequency of incorrect responses should be limited by the program. If the student is not able to provide a correct answer in 2 or 3 tries, the program should branch appropriately. If the program is "held in check" until the correct answer is given, frustration on the learner's part will result.

6. Frequent testing should be provided by the program, and record-keeping capabilities should be available. This allows the teacher to monitor progress made by the students who interact with the program.

Games and Simulations

Games and simulations are combined in this section because good simulations are in a game format. This is not to be confused with the video games currently found in arcades and in homes. The goal in those cases is to get points, the acquisition of the most possible points being the sole objective of playing the game. Good simulations have a built-in way of keeping score but the results mean more than the numerical output. The points are indicative of a degree of educational achievement that has taken place during the activity.

The subject areas benefitting from good similation software are, at present, limited. In fact, the vast majority can be labeled as exercises in economics. The titles are different and the "locale" changes, but the activity remains about the same. The student might be asked to plan a trip, or he might assume the role of an owner of a fast-food stand. In each case, however, success is measured in the degree of economic efficiency attained.

Do students enjoy simulation activities? Without a doubt, yes, if the software meets the following criteria:

1. There should be varying levels of difficulty. The

beginner must have an opportunity to gain a degree of success at a lower level. Without it, motivation will be lacking in that few people tend to be eager to try something in which they have no chance of success. On the other end of the continuum is the student who has already had some success at the lower levels of difficulty and wishes to be challenged to a greater degree. The different levels provide the potential for high motivation over an extended period of time.

2. If the program does not allow the student to select a level at which to operate, there should be an automatic increase in simulation difficulty when an increase becomes necessary. This type of program is rare, at present, but the advances being made in software development suggest it is but a matter of time. In theory, the computer will "learn" as it competes against the student in the simulation. In order to remain competitive, the student must continually develop his skills.

3. The simulation should be generally realistic. The student asked to role play will be reluctant to do so if realism is lacking. The more believable the situation, the better the simulation.

4. A high level of interest must be maintained throughout the program. Aspects of motivation have been previously addressed in this chapter.

5. The student should be able to exit (stop participation) at any time. There should also be the opportunity to reenter the program at the exit location. This is an absolute must with some software that is designed to be used over an extended period of time.

Even though some suggest it holds as much or more promise for making a significant impact on the education of youth in the public schools, simulation software is not nearly as prevalent as is the drill and practice variety. If it is well conceived, motivating students to use it will be of little concern since students enjoy learning through gaming. The present challenge lies with the software producers' abilities to create such software to be used in all subject areas.

Administrative

Administrative software includes the non-instructional software packages. They are designed to allow the user to store and retrieve information in an efficient manner, in terms of both space and time. This software may be used in the offices of the principal, media specialist, counselor, nurse, and classroom teacher. The custodian could likely find uses for these programs, too. Each of these people might use a single piece of software for a different reason. The software is labeled "administrative" because of the purpose it serves rather than the place it may be found in the school.

1. Instructions for use of the software should be clear and complete. It is assumed that most users have little computer background experience and need to be guided through the process. A menu-driven program best exemplifies this. The user is given a menu of choices from which to select and the machine does the rest. (Actually, it's not that simple, but some programs are almost this easy.)

2. A single software package should have the flexibility to serve more than one master, especially if the cost of the package exceeds the normal cost of software. A program that allows the user to store health information on each student in school should also be adaptable for the teacher who wishes to keep a record of parent conferences. Many packages allow the user to develop, from scratch, his own file to meet his own needs.

3. Ease and speed of information (data) handling is important. If it is more difficult and more time consuming to store (enter) and retrieve data using the computer than it would be the "old fashioned way," it makes little sense to use the computer. This would be too high a price to pay just to say that a school has entered the computer age. However, sometimes information storage takes a little more time than normal but yields benefits in the retrieval stage.

4. Storage capacity is important. Each software package designed for an administrative use will have limits as to the amount of information, or number of sets of information, it can hold. Future uses should also be considered when planning for data storage.

5. Security of data is vital. Most information stored in files has a degree of confidentiality about it. There must be a way of protecting the privacy of those whose records are stored. This is most often accomplished through one or more passwords. In order to access (get into) a file, the user would need to enter the proper password. This need not be a complicated matter.

6. The package should allow the user to edit the data. This includes adding entries to the file and deleting entries, also. The ability to search for a particular file, or set of files, is also a requirement.

7. There should be the capability of making a backup copy of the data disk. This is seldom a difficulty but without it problems can arise. Consider the time and effort needed to replace the data if a disk becomes damaged or lost.

8. If the program provides the opportunity to print out reports from the stored data, so much the better. This might be in the file program itself or in another package designed to be an "add-on" to the file program.

Summary

Software is abundant for a multitude of purposes in the schools. It may be used with students for drill and practice of skills already taught. It is available on a limited basis to serve as a tutorial aid with new material. Simulation activities allow the student to further develop skills in a game situation. For those needing assistance in record keeping, quality software is available.

Word processing is yet another type of software that will undoubtedly play a role in schooling, for both staff and students. The authors wrote this monograph with the assistance of two different word processing programs, using their own personal computers and dot matrix printers to do so. After much discussion they elected to forego addressing this particular use of software. It is in a category by itself in that the complexities are greater in number and so are the possibilities. An entire monograph could be devoted to the subject.

When purchasing software, it is best to purchase from a reputable supplier. The company that produces the software initially should have a reputation for quality work. As schools acquire software they will become sources of information. Various journals contain software reviews. Different localties have user groups for particular hardware (Apple users, for instance) and these people will likely have information to share concerning software for their particular machines. Some teachers and administrators will read the journals and belong to the user groups. Seek information from them before buying.

The issue of cost versus quality is an important one to consider. The most expensive is not always the best, but quality generally costs more than an inferior product. Price of software should be considered, however, after the most appropriate packages have been identified. Then, if the cost of the first choice is too high, the selection committee likely will have good second alternatives. At times, the only software available that truly meets a particular need costs more than the school people wish to spend. If it gets ruled out before the selection process begins, one of two things will happen. Either no software will be selected or the committee will select an inferior product, not realizing how inferior it is because they had nothing with which to compare it. Identifying appropriate software prior to pricing will effectively decrease the possibility of making unwise purchases.

Software selection takes time and effort on top of the knowledge and understanding of what to look for. The very best people should be assigned to do the job. The consequences of poor decisions are far too great to be left to those who fail to realize the importance of the task.

CLOSING COMMENTS

If you began with the introduction of this monograph and read each succeeding chapter prior to arriving at this summary statement, it is fair to say that technological advances have been made during this interim. It is likely some hardware has been improved through increases in memory and speed of operation while the prices have held steady or become lower. There can be no doubt that new software has been introduced, since that appears to be a daily occurrence.

Good things are happening in the schools, too. Teachers and administrative personnel are making progress. They are gradually becoming educated concerning some important uses to be made of computers. Even though this progress might be rather slow, it is progress, and patience must be exercised when encouraging educator involvement in computing.

Public schools in this country have existed for better than three centuries without computers. The "geniuses" responsible for inventing computers are products of these public schools. It is reasonable to assume schools could continue to exist without the machines being on site, but that should not be allowed to happen.

The computer has already produced so much interest in education, from within and outside the schools, that to allow the inherent opportunities to escape would do a real injustice to the curriculum. It cannot become static. It must remain dynamic. To fail to progress is to move backward. Historically, the schools have responded to the wishes of the society they serve, even when those societal expectations were unrealistic, ill-founded, or both. The present concerns for computer literacy, by whatever definition, must be answered and they will be. It behooves us all to respond in a manner that is educationally sound.

Appendix

SELECTED LIST OF SOFTWARE COMPANIES

Apple Computer Inc.
20252 Mariani Avenue
Cupertino, CA 95014

C & C Software
5713 Kentford Court
Wichita, KS 67220

George Chapas
Instructional Design Assoc.
8000 Utah Street
Merrillville, IN 46410

Compu-Tations Inc.
P.O. Box 502
Troy, MI 48099

Computer Contact Inc.
3017 E. Hill Rd., Box 7
Grand Blanc, MI 48439

Computer Stations Inc.
11610 Page Service Drive
St. Louis, MO 63146

Counterpoint Software, Inc.
4005 West 65th Street
Minneapolis, MN 55435

Creative Publications
P.O. Box 10328
Palo Alto, CA 94303

Cross Educational Software
1802 North Trenton Street
P.O. Box 1536
Ruston, LA 71270

Custom Software Inc.
Honey Creek West
3297 South 3rd Place
Terre Haute, IN 47802

Cybertronics International
Software Publishing Division
999 Mount Kemble Avenue
Morristown, NJ 07960

dilithium Press
P.O. Box 606
Beaverton, OR 97005

George Earl
1302 S. General McMullen
San Antonio, TX 78237

Essertier Software Corporation
919 Fourteenth Street
Hermosa Beach, CA 90254

Hartley Courseware Inc.
Box 431
Dimondale, MI 48821

Howard W. Sams & Co., Inc.
4300 West 62nd Street
P.O. Box 558
Indianapolis, IN 46206

Human Relations Media
175 Tompkins Avenue
Pleasantville, NY 10570

J & S Software
140 Reid Avenue
Port Washington, NY 10570

Merry Bee Communications
815 Crest Drive--Papillion
Omaha, NE 68046

Micro Learn
A Division of Micro Lab
2699 Skokie Valley Road
Highland Park, IL 60035

Milliken Publishing Company
1100 Research Boulevard
St. Louis, MO 63132

Odesta
3186 Doolittle Drive
Northbrook, IL 60062

The Professor
P.O. Box 301
Swanton, VT 05488

Programs for Learning
P.O. Box 954
New Milford, CT 06776

Quality Educational Designs
P.O. Box 12486
Portland, OR 97212

Scarborough Systems Inc.
25 North Broadway
Terrytown, NY 10591

Sensible Software
6619 Perham Drive
West Bloomfield, MI 48033

Sierra On-Line
Sierra On-Line Building
Coarsegold, CA 93614

Silicon Valley Systems
1625 El Camino Real, #4
Belmont, CA 94002

Silver Burdette Company
c/o Charles E. Richardson
7210 Mardyke Lane
Indianapolis, IN 46226

Sir-Tech Software Inc.
6 Main Street
Ogdensburg, NY 13669

Software Publishing
Corporation
1901 Landings Drive
Mt. View, CA 94043

SouthWest EdPsych
Services Inc.
6526 W. Villa Theresa
Glendale, AZ 85308

Sterling Swift Publishing Co.
7901 South IH-35
Austin, TX 78744

Terrapin Inc.
380 Green Street
Cambridge, MA 02139

Universal Systems for
Education
14901 E. Hampden Avenue
Suite 250
Aurora, CO 80014

Additional copies of this bulletin may be ordered from:

CURRICULUM RESEARCH AND DEVELOPMENT CENTER
Statesman Towers West
School of Education
Indiana State University
Terre Haute, Indiana 47809

Other bulletins currently available:

CULTURAL AWARENESS AND READING
A PAGEANT IN READING
READING CHALLENGES: NEW DIRECTIONS
THE READING ENVIRONMENT
RHYTHM, RHYME, AND REASON
READING: FOCUS FOR ENJOYMENT
READING: GREAT EXPECTATIONS

CURRENT ISSUES IN SOCIAL STUDIES EDUCATION
RE-EDUCATION IN HUMAN COMMUNICATIONS
ACTIVITIES FOR ELEMENTARY SCHOOL MATHEMATICS ENRICHMENT
ALTERNATIVES TO TESTS, MARKS, AND CLASS RANK
LEARNING ACTIVITY PACKAGES

CURRICULAR TRENDS IN INDIANA PUBLIC SCHOOLS
ENVIRONMENTAL EDUCATION IN INDIANA PUBLIC SCHOOLS
PERCEPTIONS OF FUTURE ELEMENTARY SCHOOL PRINCIPALS
THE ROLE OF CLASSROOM TEACHERS IN STUDENT SELF-CONCEPT DEVELOPMENT
THE ARENAS OF POWER: FOCUS ON SCHOOLING

CURRICULUM EVOLUTION AS PORTRAYED THROUGH OLD TEXTBOOKS
MINORITY ETHNIC EDUCATION IN INDIANA PUBLIC SCHOOLS
BLACK PEOPLE IN AMERICA: A TEACHING-LEARNING UNIT
THE TEACHER'S ROLE IN OTHERS-CONCEPT DEVELOPMENT
THE DIFFERENCE BETWEEN THORNS AND THREATS IN ELEMENTARY SCHOOL ADMINISTRATION

THE THORNS AND THREATS OF SECONDARY SCHOOL ADMINISTRATION
A DEVELOPING CURRICULUM IN ENGLISH
INDIANA HISTORY TEST: FOURTH GRADE
THE AMERICAN INDIAN: A TEACHING-LEARNING UNIT
LEADERSHIP FOR EFFECTIVE EVALUATION OF LEARNING

CURRICULUM DEVELOPMENT IN INDIANA: ANALYSIS OF THE C-1 RULE AND CAPPS
TWELVE-YEAR FOLLOW-UP STUDY: ROLE AND STATUS OF CURRICULUM WORKERS IN INDIANA
USING MEASUREMENT & EVALUATION TO PROMOTE LEARNING
SOCIALLY MAINSTREAMING HANDICAPPED STUDENTS
MEASUREMENT AND EVALUATION IN INDIANA PUBLIC SCHOOLS
IMPROVING TEST TAKING SKILLS
LEGAL ASPECTS OF PUPIL EVALUATION
CURRICULUM IN ENGLISH I: OPTIONS, GOALS, CONTENT
CURRICULUM IN ENGLISH II: PLANNING AN INTEGRATED UNIT